Please replace the *Summary of Claimed Subject Matter* section with the following amended section:

VII. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claims 31and 34 are subject of the present appeal, and recite the same invention in method and apparatus categories, respectively. ~~The subject matter of the independent claims is described with reference to the independent method claim 31.~~

The claims describe a method for mutual authentication in communications between first and second stations, such as the server 1001 and client 1003 in Figure 1. For the purpose of clarity in the following discussion, the first station is referred to as the server and the second station is referred to as the client. The invention provides for mutual authentication of the server and the client, without relying on previously distributed encryption keys and thereby reducing administrative costs. Also, the invention provides for mutual authentication without sending shared secrets, such as passwords, neither in clear text nor in encrypted text, across the communication channel. Thereby the security of the protocol is improved.

Independent claim 31 recites a method. The method includes generating and storing ephemeral session keys SRK1, SRK2, SRK3, ... (See reference numeral 1013 in Figure 1), and sets of intermediate data keys DRK1, DRK2, DRK3, ... (see reference numeral 2013 in Figure 2).

The server generates and stores a set of ephemeral session keys SRK1, SRK2, SRK3, ... (See reference numeral 1013 in Figure 1). Each of the ephemeral session keys is associated with a session key initiation interval, also called a "first lifetime LT1" (see, paragraph [0039]). Although not in claim 31, each ephemeral session key is described in the specification as having a second lifetime LT2 corresponding to the length of time that it remains in the server memory ASK 1013 before it is discarded (see, page 13, lines 16-30). The first lifetime LT1, referred to in the claims as the session key initiation interval, and its use are unique elements of the claims.

The server also generates and stores a set of intermediate data keys DRK1, DRK2, DRK3, ... The independent claim recites a single set of intermediate data keys. In the embodiment described in the application, unique sets of intermediate data keys 2013 are generated for each client session (reference numeral 2005, Figure 2).

According to the claimed method, the server responds to a request to initiate a communication session from the client by selecting the session key associated with the session key initiation interval during which the request is received (Figure 3, step 1, reference numeral 3005). Figure 1 shows requests (reference numerals 1006, 1015, 1016, 1017) made during the ninth minute, the fifth minute, the second minute, and the fifth minute, respectively. The requests fall within specific session key initiation intervals. The server selects the session keys in response to the requests, that are associated with the session key initiation interval in which the requests were made, for use in initiation of the requested session.

The associated session key is then sent to the client (Figure 3, step 2, reference numeral 3006), which returns a message carrying a digital identifier, such as a user name, encrypted using the associated session key (Figure 3, steps 3-4 (reference numeral 3007, 3008). The server is able to verify receipt of the associated session key and identify the client after decrypting the digital identifier carried in this message, and matching it with account records.

The server and client execute first and second sets of exchanges. The first set of exchanges results in delivery of an intermediate data key DRK1, or more generally DRK(n-1) to the client from the server (step numbers 4-5 (reference numerals 8000-8008) in Figure 8A). As stated in the specification, a larger number of exchanges utilized in the first set of exchanges results in greater security for the algorithm. However, for simplicity, we can refer to the results of the first set of exchanges as delivery of the first intermediate data key DRK1. The claims do not require the use of more than one intermediate data key in the first set of exchanges.

The second set of exchanges accomplishes mutual authentication using intermediate data keys up to DRK2, or more generally DRK(n), a first shared secret and a second shared secret (step numbers 6-9 (reference numerals 9000-8008) in Figure 8A). The first shared secret can be a client password (h-u-password). The second shared secret can be a server password (h-s-password).

Using the second set of exchanges, the server is able to authenticate the client and the client is able to authenticate the server by a process in which the first and second shared secrets are used for encrypting intermediate data keys. The shared secrets are not delivered across the communication channel. One way of using the shared secrets for the encryption of the intermediate data keys, for example, is shown at steps to 6-7 (reference numerals 8000-8008) on Figure 8A. In steps 6-7 (reference numerals 8000-8008) on Figure 8A, the server sends an intermediate data key DRK2, or more generally DRK(n), encrypted using intermediate data key DRK1, or more generally DRK(n-1), to the client. The client seeds a veiling algorithm (*e.g.*, Byte-VU, page 20, line 26 through page 21, line 7) using the client password (first shared secret) that is used in produce an encrypted version of intermediate data key DRK(n).

The hashed version of intermediate data key DRK(n), labeled h-DRK2 in Figure 8A, is returned to the server after encryption using the same intermediate data key DRK(n).

Independent claim 34 recites a data processing apparatus, parallel in substance with independent claim 31, which performs the server side functions described above. It recites a data processor (e.g. server 1001), that includes a communication interface (e.g. WWW server 1002), memory storing instructions that include logic to provide for mutual authentication. The logic includes element 1005 of Fig. 1 for generating and storing a set of ephemeral session keys, and element 2005 of Fig. 2 for generating and storing a set of intermediate data keys. In addition, the logic in claim 34 includes the elements 3005-3016 shown in Fig. 3, executed on the server side, as detailed above in connection with the discussion of claim 31.

Therefore, the present invention provides a protocol for mutual authentication that is completely different than any in the prior art. Using this protocol, the first and second shared secrets need not be exchanged on the communication channel during the session. Also, using this protocol, no encryption keys need to be exchanged in advance of the session. Nonetheless, the server and client are mutually authenticated in a highly secure fashion.

///